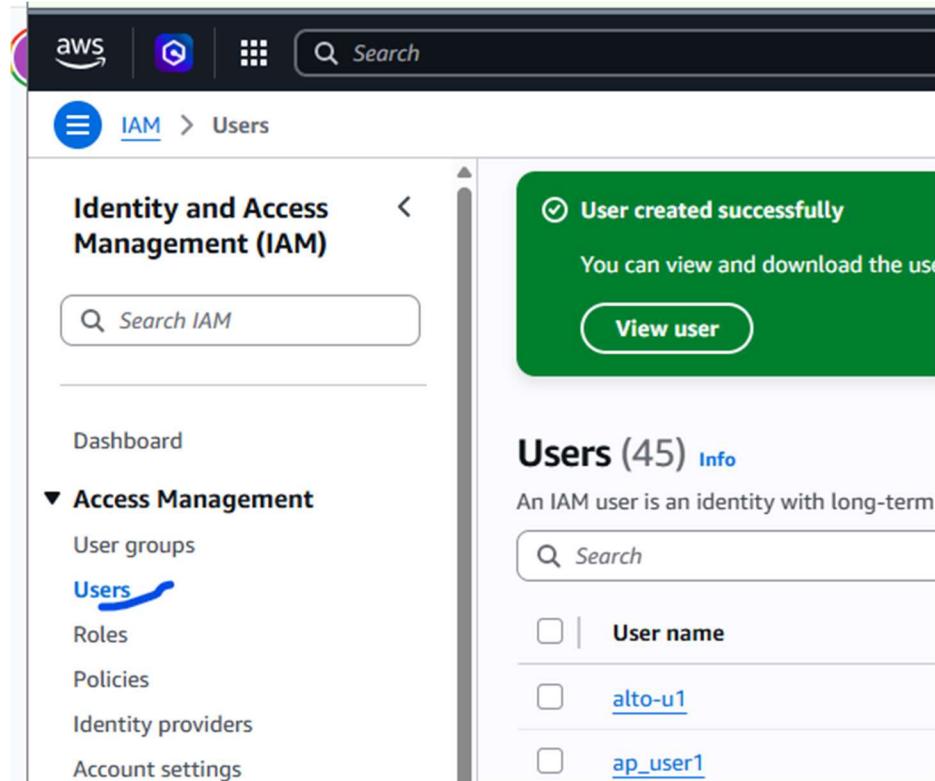


Create IAM user

Steps

1. Create iam group/user with ec2 full access
2. Create access key and secret

1. Create iam group/user with ec2 full access



Click on user and then on the right side , click on “Create user”

Step 1
 Specify user details
 Step 2
 Set permissions
 Step 3
 Review and create

Specify user details

User details

User name

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + . _ - (hyphen)

Provide user access to the AWS Management Console - optional
In addition to console access, users with SigninLocalDevelopmentAccess permissions can use the same console credentials for programmatic access without the need for access keys.

If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can generate them after you create this IAM user.
[Learn more](#)

Cancel **Next**

Click Next.

Step 1
 Specify user details
 Set permissions
 Step 3
 Review and create

Set permissions
Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job function. [Learn more](#)

Permissions options

Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.

Attach policies directly
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

User groups (5)

Group name	Users	Attached policies	Created
Admin-Group-A2	1	AdministratorAccess	2025-04-04 (10 months ago)
b06-q1	1	AmazonEC2ReadOnlyAccess	2025-08-18 (6 months ago)
g1	0	vpc-full-access-may25	2025-05-11 (9 months ago)
group1	1	AmazonEC2FullAccess	2025-09-08 (5 months ago)
specnt-tf	31	-	2025-01-31 (1 year ago)

Set permissions boundary - optional

Cancel **Previous** **Next**

Click on “Attach policies directly”

And in the permission search for “ec2full”

Step 1 Specify user details

Step 2 Set permissions

Step 3 Review and create

Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Permissions options

- Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.
- Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.
- Attach policies directly
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

Permissions policies (1590)

Choose one or more policies to attach to your new user.

Filter by Type		
<input type="text" value="ec2full"/>	All types	1 match
<input type="checkbox"/> Policy name A	Type	Attached entities
<input type="checkbox"/> AmazonEC2FullAccess	AWS managed	6

[Create policy](#)

▶ Set permissions boundary - *optional*

Cancel Previous Next

Put a tick mark and click on next.

sers > Create user

Step 1 Specify user details

Step 2 Set permissions

Step 3 Review and create

Review and create

Review your choices. After you create the user, you can view and download the autogenerated password, if enabled.

User details

User name tf-user01	Console password type None	Require password reset No
------------------------	-------------------------------	------------------------------

Permissions summary

Name A	Type	Used as
AmazonEC2FullAccess	AWS managed	Permissions policy

Tags - optional

Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

Cancel Previous [Create user](#)

Click on Create user.

2. Create access key and secret

Click on the user.

The screenshot shows the AWS IAM User Details page for a user named 'tf-user01'. The left sidebar includes links for Identity and Access Management (IAM), Dashboard, Access Management (User groups, Roles, Policies, Identity providers, Account settings, Root access management, Temporary delegation requests), and Access reports (Access Analyzer, Resource analysis). The main content area has tabs for Summary, Permissions, Groups, Tags, Security credentials, and Last Accessed. The Summary tab displays the ARN (arn:aws:iam::443370714870:user/tf-user01), Console access status (Disabled), and Last console sign-in (February 20, 2026, 15:23 (UTC+05:30)). The Permissions tab shows one policy attached: 'AmazonEC2FullAccess' (AWS managed). A blue arrow points to the 'Create access key' link in the top right corner of the Security credentials section.

Click on “Security credentials”

The screenshot shows the AWS IAM User Security Credentials page for 'tf-user01'. The left sidebar lists IAM users and other IAM-related links. The main content area has tabs for Security credentials, Permissions, Groups, Tags, and Last Accessed. The Security credentials tab contains sections for Console sign-in (Console sign-in link: https://443370714870.signin.aws.amazon.com/console, Console password: Not enabled) and Multi-factor authentication (MFA) (0). It also includes a note about MFA devices and a 'Create MFA device' button. The Access keys (0) section notes that no access keys are present and provides a 'Create access key' button. A blue arrow points to the 'Create access key' button in the Access keys section.

Click on “Create access key”

Step 1
Access key best practices & alternatives

Step 2 - optional
 Set description tag

Step 3
 Retrieve access keys

Access key best practices & alternatives Info

Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alternatives.

Use case



Command Line Interface (CLI)

You plan to use this access key to enable the AWS CLI to access your AWS account.



Local code

You plan to use this access key to enable application code in a local development environment to access your AWS account.



Application running on an AWS compute service

You plan to use this access key to enable application code running on an AWS compute service like Amazon EC2, Amazon ECS, or AWS Lambda to access your AWS account.



Third-party service

You plan to use this access key to enable access for a third-party application or service that monitors or manages your AWS resources.



Application running outside AWS

You plan to use this access key to authenticate workloads running in your data center or other infrastructure outside of AWS that needs to access your AWS resources.



Other

Your use case is not listed here.

[Cancel](#)

[Next](#)

Select the first option “command line interface”

Click on next

Create access key

Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alternatives.

tag

keys

Use case



Command Line Interface (CLI)

You plan to use this access key to enable the AWS CLI to access your AWS account.



Local code

You plan to use this access key to enable application code in a local development environment to access your AWS account.



Application running on an AWS compute service

You plan to use this access key to enable application code running on an AWS compute service like Amazon EC2, Amazon ECS, or AWS Lambda to access your AWS account.



Third-party service

You plan to use this access key to enable access for a third-party application or service that monitors or manages your AWS resources.



Application running outside AWS

You plan to use this access key to authenticate workloads running in your data center or other infrastructure outside of AWS that needs to access your AWS resources.



Other

Your use case is not listed here.

Alternatives recommended

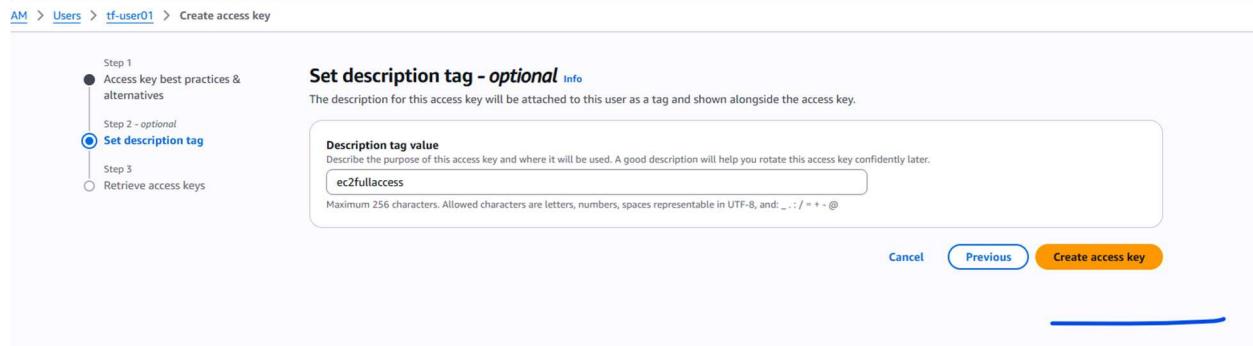
- Use AWS CLI V2 and the aws login command to use your existing console credentials in the CLI. [Learn more ↗](#)
- Use AWS CloudShell, a browser-based CLI, to run commands. [Learn more ↗](#)

Confirmation

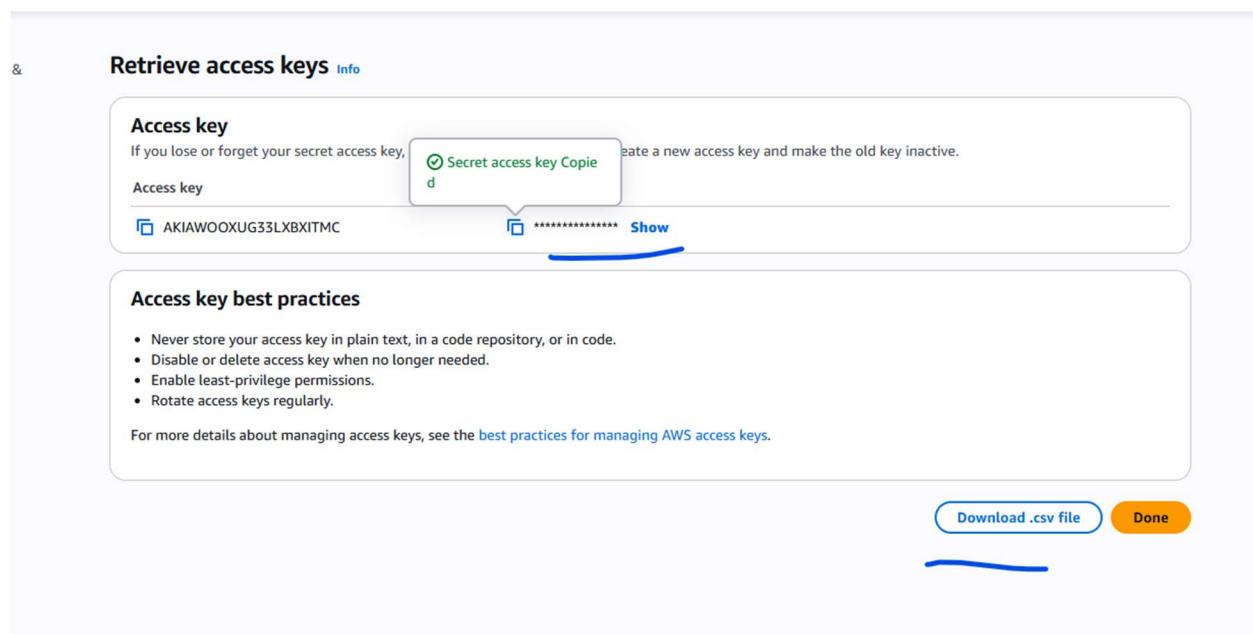
I understand the above recommendation and want to proceed to create an access key.

[Cancel](#)

[Next](#)



Click on “create access key”.



Copy the secret and save it , as it would NOT be available again

Or download the .csv file